Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1-3. (Cancelled)

4. (Currently Amended) A ~~Java~~JAVA^{TM} virtual machine residing on a computing apparatus and operating in a ~~Java~~JAVA^{TM} computing environment, said ~~Java~~JAVA^{TM} virtual machine capable of determining a string representation associated with a ~~Java~~JAVA^{TM} object, wherein said virtual machine determines said string representation of said ~~Java~~JAVA^{TM} object without invoking a ~~Java~~JAVA^{TM} “to_string” method[[.]], wherein said virtual machine is capable of:

pushing a reference to said JAVA^{TM} object on an execution stack;

popping said reference to said JAVA^{TM} object from said execution stack;

determining a string representation of a field associated with said JAVA^{TM} object by accessing said JAVA^{TM} object using said reference; and

pushing a reference to said string representation of said field on top of said execution stack.

5. (Currently Amended) A ~~Java~~JAVA^{TM} virtual machine as recited in claim 4, wherein said ~~Java~~JAVA^{TM} virtual machine executes ~~an inventive~~ a ~~Java~~JAVA^{TM} Bytecode instruction, said ~~inventive Java~~JAVA^{TM} Bytecode instruction operating to determine said string representation associated with said ~~Java~~JAVA^{TM} object; thereby allowing said string representation to be determined without invoking a ~~Java~~JAVA^{TM} method.

6. (Cancelled)

7. (Currently Amended) A ~~Java~~JAVA™ virtual machine as recited in claim 5, wherein said ~~Java~~JAVA™ virtual machine operates in an embedded system.

8. (Currently Amended) In a ~~Java~~JAVA™ computing environment, a method of retrieving by a virtual machine a string representation for a ~~Java~~JAVA™ object, said virtual machine residing on a computing apparatus, said method comprising:

receiving a~~n inventive Java~~JAVA™ Bytecode instruction in a stream of ~~Java~~JAVA™ Bytecodes suitable for execution by a virtual machine operating in said ~~Java~~JAVA™ computing environment~~, and~~;

pushing a reference to said JAVA™ object on an execution stack;

popping said reference to said JAVA™ object from said execution stack;

determining a string representation of a field associated with said JAVA™ object by accessing said JAVA™ object using said reference; and

pushing a reference to said string representation of said field on top of said execution stack.

wherein said ~~inventive Java~~JAVA™ Bytecode instruction operates to determine said string representation associated with said ~~Java~~JAVA™ object; thereby allowing said string representation to be determined without invoking a ~~Java~~JAVA™ method.

9. (Cancelled)

10. (Cancelled)

11. (Currently Amended) A method as recited in claim 8, wherein said pushing of a reference to said ~~Java~~JAVA^{TM} object is performed by execution of a ~~Java~~JAVA^{TM} Aload execution.

12. (Currently Amended) A method as recited in claim 11, wherein said method is performed by a virtual machine.

13. (Currently Amended) A method as recited in claim 12, wherein said virtual machine is operating in an embedded system.

14. (Currently Amended) A computer readable ~~media~~ medium including computer program code for retrieving a string representation for a ~~Java~~JAVA^{TM} object, said computer readable ~~media~~ medium comprising:

computer program code for receiving ~~an inventive~~ a ~~Java~~JAVA^{TM} Bytecode instruction in a stream of ~~Java~~JAVA^{TM} Bytecodes suitable for execution by a virtual machine operating in ~~said~~ a ~~Java~~JAVA^{TM} computing environment, and

wherein said ~~inventive Java~~JAVA^{TM} Bytecode instruction operates to determine said string representation associated with said ~~Java~~JAVA^{TM} object; thereby allowing said string representation to be determined without invoking a ~~Java~~JAVA^{TM} method.

15. (Currently Amended) A computer readable ~~media~~ medium as recited in claim 14, wherein said computer readable ~~media~~ medium further comprises:

computer program code for popping a reference to a ~~Java~~JAVA^{TM} object from an execution stack;

computer program code for determining a string representation of a field associated with said ~~Java~~JAVA^{TM} object; and

computer program code for pushing a reference to said string representation of said field on top of said execution stack.

16. (Cancelled)

17. (Currently Amended) A computer readable ~~media~~ medium as recited in claim 15, wherein said computer program code ~~for pushing said reference is performed by executing~~ comprises a ~~Java~~JAVA™ Aload instruction that when executed performs the pushing of said reference.

18. (Currently Amended) A computer readable ~~media~~ medium as recited in claim 17, wherein said computer readable media is read by a ~~Java~~JAVA™ virtual machine.

19. (Currently Amended) A computer readable ~~media~~ medium as recited in claim 18, wherein said virtual machine is operating in an embedded system.

20. (New) A computer system for retrieving a string representation for a JAVA™ object in a JAVA™ computing environment, said computer system capable of operating to:

receive a JAVA™ Bytecode instruction in a stream of JAVA™ Bytecodes suitable for execution by a virtual machine operating in said JAVA™ computing environment;

pushing a reference to said JAVA™ object on an execution stack;

popping said reference to said JAVA™ object from said execution stack;

determining a string representation of a field associated with said JAVA™ object by accessing said JAVA™ object using said reference;

push a reference to said string representation of said field on top of said execution stack; and wherein said JAVA^{TM} Bytecode instruction operates to determine said string representation associated with said JAVA^{TM} object; thereby allowing said string representation to be determined without invoking a JAVA^{TM} method.

21. (New) A computer system as recited in claim 20, wherein said pushing of a reference to said JAVA^{TM} object is performed by execution of a JAVA^{TM} Aload bytecode.

22. (New) A computer system as recited in claim 21, wherein said virtual machine operates in an embedded system.